CLAIMS

What is claimed is:

1.   A method for marking a processing stack with signatures to indicate which

portions of the stack were utilized by which software code modules, said

5   method comprising the step of:

assigning unique module identifier values to a plurality of code

modules; and

upon execution of said code modules, pushing onto a processing

stack said unique module identifiers within stack frames allocated to said

10   code modules.

2.   The method as set forth in Claim 1 further comprising the steps of:

generating an instance number for each instantiation of a code module;

and

pushing onto said processing stack said instance numbers associated

15   with said unique module identifier values.

3.   The method as set forth in Claim 1 further comprising the step of pushing onto

said stack an entry/exit indicator associated with said unique module

identifier.

4.   The method as set forth in Claim 1 further comprising the step of inserting

20   stack signature marking software segments into application source code, said

insertion step being performed prior to compilation of said application source

code.

5.   The method as set forth in Claim 4 further comprising the step of providing a

global control which indicates all application source code modules are to have

stack signature marking software segments inserted into them during a given

compilation job.

5   6.   The method as set forth in Claim 4 further comprising the step of providing a

selective control which indicates only certain application source code modules

are to have stack signature marking software segments inserted into them

during a given compilation job.

7.   A computer readable medium encoded with software for marking a processing

10   stack with signatures to indicate which portions of the stack were utilized by

which application code modules, said software causing a processor to perform

the steps of:

assigning unique module identifier values to a plurality of code

modules; and

15   upon execution of said code modules, pushing onto a processing

stack said unique module identifiers within stack frames allocated to said

code modules.

8.   The computer readable medium as set forth in Claim 7 further comprising

software to perform the steps of:

20   generating an instance number for each instantiation of a code module;

and

pushing onto said processing stack said instance numbers associated

with said unique module identifier values.

9.   The computer readable medium as set forth in Claim 7 further comprising

software for performing the step of pushing onto said stack an entry/exit

indicator associated with said unique module identifier.

5   10.   The computer readable medium as set forth in Claim 7 further comprising

software for performing the step of inserting stack signature marking software

segments into application source code, said insertion step being performed

prior to compilation of said application source code.

11.   The computer readable medium as set forth in Claim 10 further comprising

10   software for performing the step of providing a global control which indicates

all application source code modules are to have stack signature marking

software segments inserted into them during a given compilation job.

12.   The computer readable medium as set forth in Claim 10 further comprising

software for performing the step of providing a selective control which

15   indicates only certain application source code modules are to have stack

signature marking software segments inserted into them during a given

compilation job.

13.   A system for inserting stack signature marking code segments into application

software modules prior to compilation, said system cooperating with a

20   compiler and comprising:

a control means operable by a user to indicate whether or not to insert

stack signature marking code segments into application software modules; and

a code insertion means which, responsive to the operation of the control means, searches for entry points and exits points in application software modules and inserts stack signature marking code segments following each entry point and prior to each exit point into said application

5 software modules.

14. The system of Claim 14 wherein said control means comprises a global control means for indicating insertion of stack signature marking code segments are to be inserted into all application software modules to be compiled.

10 15. The system of Claim 15 wherein said control means comprises a selective control means for indicating specific applications software modules or groups of application software modules into which stack signature marking code segments are to be inserted.